Claims

1. Signal delaying device (1) for the dynamic delaying of a digitally sampled input signal with a memory element (2) and a series connected interpolation element (3), wherein, a register (30), which can be connected to the output side of the interpolation element (3) for the intermediate storage of at least one sampled value ($S_{in}(k)$) of the input signal, is arranged in parallel to the memory element (2).

2. Signal delaying device according to claim 1, **characterised in that**
a marking device (31) is provided, which, after a sampled value ($S_{in}(k)$) of the input signal has been placed in intermediate storage in the register (30), adds a marking to the next sampled value ($S_{in}(k+1)$) of the input signal stored in the memory element (2).

3. Signal delaying device according to claim 2, **characterised in that**
the interpolation element (3) checks whether the marking has arrived at the output of the memory element (2), and following this, reads out a sampled value ($x(k)$) from the memory element (2) and also a sampled value from the register (30).

4. Signal delaying device according to any one of claims 1 to 3,
characterised in that

the interpolation element (3) comprises a polyphase filter (5).

5. Signal delaying device according to claim 4, **characterised in that** the interpolation element (3) comprises a half-band filter (4), which is arranged between the memory element (2) and the register on one side, and the polyphase filter on the other side.

6. Method for the dynamic delaying of a digitally sampled input signal with the following procedural stages:
- storage of the sampled values of the input signal in a memory element (2),
- reading out of the sampled values ($S_{in}(k)$) from the memory element (2),
- interpolation of the sampled values ($x(k)$) read out from the memory element (2),
wherein
- whenever the range (19) defined by two successive sampled values ($x(k-4)$, $x(k-3)$) is neither undercut nor exceeded in the interpolation, one sampled value ($S_{in}(k)$) is placed into the memory element (2) and one sampled value ($x(k)$) is read out from the memory element (2),
- whenever the range (20) defined by two successive sampled values ($x(k-4)$, $x(k-3)$) is exceeded in the interpolation, no new sampled value ($x(k)$) is read out from the memory element (2),
- before the range (21) defined by two successive sampled values ($x(k-4)$, $x(k-3)$) is undercut in the interpolation, a sampled value ($S_{in}(k)$) of the input signal is placed in intermediate storage in a

register (30) arranged in parallel to the memory element (2), the next sampled value (Sin(k+1)) of the input signal stored in the memory element (2) is marked, and a sampled value from the memory element (2) and also the sampled value placed in intermediate storage in the register (30) are read out, whenever the marked sampled value arrives at the output of the memory element (2).

7. Method according to claim 6,
characterised in that
the range (20) defined by two successive sampled values (x(k-4), x(k-3)) is exceeded, if at least two interpolation values ($S_{out}(k-3)$, $S_{out}(k-2)$) produced by the interpolation fall within this range (20).

8. Method according to claim 6 or 7,
characterised in that
the range (21) defined by two successive sampled values (x(k-4), x(k-3)) is undercut in the interpolation, if no interpolation value produced by the interpolation falls within this range.

9. Method according to any one of claims 6 to 8,
characterised in that
storage in the memory element (2) takes place by means of a write pointer, and reading out from the memory element (2) takes place by means of a read pointer, wherein the write pointer and the read pointer in each case point towards a given memory cell of the memory element,
wherein the write pointer and also the read pointer are adjusted if the range (19) defined by two

successive sampled values (x(k-4), x(k-3)) is neither undercut nor exceeded in the interpolation.

10. Method according to claim 9,
characterised in that
only the write pointer but not the read pointer is adjusted, if the range (20) defined by two successive sampled values (x(k-4), x(k-3)) is exceeded in the interpolation.

11. Method according to claim 9 or 10,
characterised in that
only the read pointer but not the write pointer is adjusted, if a sampled value is stored in the register (30).

12. Method according to any one of claims 9 to 11,
characterised in that
both the write pointer and also the read pointer are adjusted, if a sampled value is read out from the register (30).